Introduction

Climate science is a Big Data domain that is experiencing unprecedented growth. In our efforts to address Big Data, we're introducing MERRA Analytic Services (MAS), which enables MapReduce analytics over NASA's Modern-Era Retrospective Analysis for Research and Applications (MERRA) data collection. The MERRA reanalysis integrates observational data with numerical models to produce a global temporally and spatially consistent synthesis of 26 key climate variables. It represents a type of data product that is of growing importance to scientists doing climate change research and a wide range of decision support applications. MAS combines high-performance MapReduce-based analytics with scalable data management and delivers capabilities through the Climate Data Services Application Programming Interface (CDS API).

See http://gmao.gsfc.nasa.gov/merra/ for more MERRA details.

Input-Output-Format

MAS input parameters are strings, but are converted internally according to the types provided below. MAS outputs are delivered in NetCDF format http://www.unidata.ucar.edu/software/netcdf/

Credits

Utility	Description
Cloudera Hadoop	Open source Hadoop platform
NetCDF for Java	NetCDF for Java Library along with GRIB/Radar/HDF decoders
Apache Commons	Reusable Java components from the Apache Foundation
HTTPClient	Support for HTTP Protocol
Maven	Software project management and comprehension tool

MAS-Method-Overview:

MERRA Analytic Service (MAS) methods are composed of basic utilities and extended utilities. Basic utilities are designed for the CDS API Web Service calls, or Python scripts and applications. The web service calls are described below with sample invocations for CDS API users.

Extended utilities, which are convenience methods that weave together multiple tasks, are accessible to Python scripts and applications only.

MAS-Basic-Utilities:

Basic utilities are designed for the CDS API Web Service calls, or Python scripts and applications.

Order

Request subsetted NetCDF data as constrained by MERRA variable, operation, time span, bounding box, and vertical level (asynchronous).

MAS orders are comprised of the parameters listed below. An order can be thought of as a 'service request' for MERRA data that is based on a specific *variable* (e.g., T for temperature). The group of available variables differs across collections and may be identified via the MERRA specification. The remaining required parameters basically define the operation, time span, spatial extents, and vertical extents associated with the order. Any variable within a collection can be operated on, and any operation in the list below is supported for each variable.

See http://gmao.gsfc.nasa.gov/research/merra/MERRA_FileSpec_DRAFT_09_02_2008.pdf for detailed descriptions of the MERRA collections and their respective variables (i.e., variable_list).

Service-Requests:

Service Request	Service	Description
GetVariableByCollection_Operation_TimeRange_SpatialExtent_VerticalExtent	MAS	Query MERRA data based on a single variable (e.g., T, or SLP)

MERRA Data Collections currently supported by MAS:

Collection	Description
instM_3d_ana_Np *	Analyzed State, Meteorology Instantaneous Monthly (p-coord, 2/3x1/2L42)
instM_3d_asm_Cp *	IAU State, Meteorology Instantaneous Monthly (p-coord, 1.25x1.25L42)
tavg1_2d_lnd_Nx	Land related surface quantities

** Collections starting with 'instM' are monthly means files, which are not directly referenced in the MERRA specification. Instead, inspect 'inst6_3d_ana_Np' in place of 'instM_3d_ana_Np' and 'inst3_3d_asm_Cp' in place of 'instM_3d_asm_Cp' when identifying MERRA variables of interest.*

Operations:

Operation	Description
avg	Calculate average value(s)
count	Calculate overall number of data points
sum	Calculate sum of all data points
min	Calculate minimum value(s)
max	Calculate maximum value(s)
var	Calculate variance across value(s)
anomaly	Calculate anomaly across value(s)

Required-Parameters:

Parameter	Description	Sample Value(s)	Range [Format]	Units
service_request *	Service request	See Range ->	GetVariableByCollection_Operation_TimeRange_SpatialExtent_VerticalExtent	String
variable_list *	The specific MERRA variable of interest	T, H, V, PS...	varies according to collection	String
collection	MERRA collection to process over	instM_3d_ana_Np	instM_3d_ana_Np, instM_3d_asm_Cp, tavg1_2d_lnd_Nx	String
operation	The operation to perform	avg	avg,min,max,sum,count,var	String
start_date	The starting date to process from	20110201	>= 197901 [yyyyMM, yyyyMMdd, yyyy-MM-dd'T'HH:mm:ss, yyyyMMdd'_'HHmmss]	String
end_date	The ending date to process to	20110630	<= 201312 [yyyyMM, yyyyMMdd, yyyy-MM-dd'T'HH:mm:ss, yyyyMMdd'_'HHmmss]	String
min_lon	Minimum longitude - degrees_east	-180.0	>= -180.0	float
min_lat	Minimum latitude - degrees_north	-90.0	>= -90.0	float
max_lon	Maximum longitude - degrees_east	180.0	<= 180.0	float
max_lat	Maximum latitude - degrees_north	90.0	<= 90.0	float
start_level	Start vertical level	1	>= 1	float
end_level	End vertical level	42	<=42	float

** The combination of the service_request and variable_list (e.g., T for temperature) indicates which MERRA variable to process within the context of the other parameters. Currently, the only service_request supported by 'order' is GetVariableByCollection_Operation_TimeRange_SpatialExtent_VerticalExtent, which also requires an operation (e.g., avg), time range, and geographical constraints.*

Optional-Parameters:

Parameter	Description	Sample Value(s)	Units
period	Time span(s) to perform an operation	For a year's worth of monthly means files (12), a period of (3) would return (4) sets of results simulating seasonal averages for example. If no period is provided, the operation is performed across the entire time span, returning a single set of results.	String
job_name	Name of Hadoop job	User-definable name that can be handy for debugging jobs on the server side (alphanumeric characters only)	String

Example Web Service Request:

```
http://skyportal.sci.gsfc.nasa.gov/cds/mas/order.php?job_name=avg_T_instM_3d_ana_Np&collection=instM_3d_ana_Np&operation=avg&serv
```

Example Web Service XML Response:

```
<Response>
    <sessionId>37734A627B5E7E74AAB8374586D3D018</sessionId>
    <sessionStatus>Received</sessionStatus>
    <sessionStatusDetail>Map Reduce job submitted at Tue Dec 10 16:29:57 EST 2013</sessionStatusDetail>
</Response>
```

Example Python Request:

```
north_american_parms = "&job_name=simple_end_to_end_use_case_na_avg_temp_2011-2012_instM_3d_ana_Np&collection=instM_3d_ana_Np&ser
cds_lib.order("MAS", "GetVariableByCollection_Operation_TimeRange_SpatialExtent_VerticalExtent", north_american_parms)
```

Future MERRA Data Collections to be supported by MAS:

Collection	Description
inst1_2d_int_Nx	Vertical integrals of quantities
inst6_3d_ana_Np	Analyzed fields at pressure levels
inst6_3d_ana_Nv	Analyzed fields on model layers
inst3_3d_asm_Cp	Basic assimilated fields from IAU corrector
tavg3_3d_cld_Cp	Upper-air cloud related diagnostics
tavg3_3d_mst_Cp	Upper-air diagnostics from moist processes
tavg3_3d_odt_Cp	Upper-air ozone tendencies by process
tavg3_3d_rad_Cp	Upper-air diagnostics from radiation
tavg3_3d_tdt_Cp	Upper-air temperature tendencies by process
tavg3_3d_trb_Cp	Upper-air diagnostics from turbulence
tavg3_3d_udt_Cp	Upper-air wind tendencies by process
tavg3_3d_qdt_Cp	Upper-air humidity tendencies by process
tavg1_2d_flx_Nx	Surface turbulent fluxes and related quantities

tavg1_2d_int_Nx	Vertical integrals of tendencies
tavg1_2d_rad_Nx	Surface and TOA radiative fluxes
tavg1_2d_slv_Nx	Single-level atmospheric state variables
tavg3_3d_chm_Fe	Chemistry related 3-D at model layer edges
tavg3_3d_chm_Fv	Chemistry related 3-D at model layer centers
tavg3_2d_chm_Fx	Chemistry related 2-DSingle-level
tavg3_3d_chm_Ne	Accumulated transport fields at edges
tavg3_3d_chm_Nv	Accumulated transport fields at layers

Status

Track progress of service activity.

Example Web Service Request:

```
http://skyportal.sci.gsfc.nasa.gov/cds/mas/status.php?session_id=37734A627B5E7E74AAB8374586D3D018
```

Example Web Service XML Response:

```
<Response>
    <sessionId>37734A627B5E7E74AAB8374586D3D018</sessionId>
    <sessionStatus>Completed</sessionStatus>
    <sessionStatusDetail>Map Reduce job completed successfully prior to Tue Dec 10 17:48:02 EST 2013</sessionStatusDetail>
</Response>
```

Example Python Request:

```
response = cds_lib.status("MAS", "37734A627B5E7E74AAB8374586D3D018")
status = cds_lib.getElement(response, "sessionStatus")
```

Download

Retrieve a Dissemination Information Package (DIP), which for MERRA/AS is a subsetted, aggregated NetCDF file.

Example Web Service Request:

```
http://skyportal.sci.gsfc.nasa.gov/cds/mas/download.php?session_id=37734A627B5E7E74AAB8374586D3D018
```

Example Web Service XML Response:

```
Payload is delivered via browser
```

Example Python Request:

```
cds_lib.download("MAS", "37734A627B5E7E74AAB8374586D3D018", "/tmp/simple_end_to_end_use_case")
```

MAS-Extended-Utilities:

Extended utilities, which are convenience methods that weave together multiple tasks, are accessible to Python scripts and applications. Note that the last parameter in each method is the local destination for the task response.

Avg

Calculate an average and download the result to the destination directory.

Executes a series of basic CDS API utilities to place an order, check status until complete and download the results to the destination.

Arguments:

```
service: The name of the target service.  For example:  MAS
parameters:  request-specific parameter list (method-dependent '&'-separated list of parameters to suffix URL with):
    see order
destination: download_folder
```

Returns:

```
Aggregate NetCDF file in destination directory
```

Example Python Request:

```
cds_lib.avg("MAS", "&job_name=avg_temp_2011-2012_instM_3d_ana_Np&collection=instM_3d_ana_Np&variable_list=T&start_date=201101&end
```

Sum

Calculate a summary and download the result to the destination directory.

Executes a series of basic CDS API utilities to place an order, check status until complete and download the results to the destination.

Arguments:

```
            service: The name of the target service.  For example:  MAS
            parameters:  request-specific parameter list (method-dependent '&'-separated list of parameters to suffix URL with):
                see order
            destination: download_folder
```

Returns:

```
            Aggregate NetCDF file in destination directory
```

Example Python Request:

```
        cds_lib.sum("MAS", "&job_name=avg_sum_2011-2012_instM_3d_ana_Np&collection=instM_3d_ana_Np&variable_list=T&start_date=201101&end_
```

Max

Calculate a maximum and download the result to the destination directory.

Executes a series of basic CDS API utilities to place an order, check status until complete and download the results to the destination.

Arguments:

```
            service: The name of the target service.  For example:  MAS
            parameters:  request-specific parameter list (method-dependent '&'-separated list of parameters to suffix URL with):
                see order
            destination: download_folder
```

Returns:

```
            Aggregate NetCDF file in destination directory
```

Example Python Request:

```
        cds_lib.max("MAS", "&job_max=avg_max_2011-2012_instM_3d_ana_Np&collection=instM_3d_ana_Np&variable_list=T&start_date=201101&end_d
```

Min

Calculate a minimum and download the result to the destination directory.

Executes a series of basic CDS API utilities to place an order, check status until complete and download the results to the destination.

Arguments:

```
            service: The name of the target service.  For example:  MAS
            parameters:  request-specific parameter list (method-dependent '&'-separated list of parameters to suffix URL with):
                see order
            destination: download_folder
```

Returns:

```
            Aggregate NetCDF file in destination directory
```

Example Python Request:

```
        cds_lib.min("MAS", "&job_min=avg_min_2011-2012_instM_3d_ana_Np&collection=instM_3d_ana_Np&variable_list=T&start_date=201101&end_d
```

Var

Calculate a variance and download the result to the destination directory. Variance measures how far a set of numbers is spread out. (A variance of zero indicates that all the values are identical.) Variance is always non-negative. A small variance indicates that the data points tend to be very close to the mean (expected value) and hence to each other, while a high variance indicates that the data points are very spread out from the mean and from each other.

Executes a series of basic CDS API utilities to place an order, check status until complete and download the results to the destination.

Arguments:

```
            service: The name of the target service.  For example:  MAS
            parameters:  request-specific parameter list (method-dependent '&'-separated list of parameters to suffix URL with):
                see order
            destination: download_folder
```

Returns:

```
            Aggregate NetCDF file in destination directory
```

Example Python Request:

```
        cds_lib.var("MAS", "&job_name=avg_var_2011-2012_instM_3d_ana_Np&collection=instM_3d_ana_Np&variable_list=T&start_date=201101&end_
```

Count

Calculate an overall count and download the result to the destination directory.

Executes a series of basic CDS API utilities to place an order, check status until complete and download the results to the destination.

Arguments:

```
            service: The name of the target service.  For example:  MAS
            parameters:  request-specific parameter list (method-dependent '&'-separated list of parameters to suffix URL with):
                see order
            destination: download_folder
```

Returns:

```
            Aggregate NetCDF file in destination directory
```

Example Python Request:

```
        cds_lib.count("MAS", "&job_name=avg_count_2011-2012_instM_3d_ana_Np&collection=instM_3d_ana_Np&variable_list=T&start_date=201101&
```

Anomaly

Calculate the difference between the overall time span average and each data point and download the result to the destination directory.

Executes a series of basic CDS API utilities to place an order, check status until complete and download the results to the destination.

Arguments:

```
            service: The name of the target service.  For example:  MAS
            parameters:  request-specific parameter list (method-dependent '&'-separated list of parameters to suffix URL with):
                see order
            destination: download_folder
```

Returns:

```
            Aggregate NetCDF file in destination directory
```

Example Python Request:

```
        cds_lib.anomaly("MAS", "&job_min=avg_min_2011-2012_instM_3d_ana_Np&collection=instM_3d_ana_Np&variable_list=T&start_date=201101&e
```